

10/590397
IAP9 Rec'd PCT/PTO 21 AUG 2006

中华人民共和国国家知识产权局
STATE INTELLECTUAL PROPERTY OFFICE
OF THE PEOPLE'S REPUBLIC OF CHINA



证 明

CERTIFICATE

本证明之附件是向中国专利局作为受理局提交的下列国际申请副本
THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY OF THE BELOW
IDENTIFIED INTERNATIONAL APPLICATION THAT WAS FILED WITH THE
CHINESE PATENT OFFICE AS RECEIVING OFFICE

国际申请号: PCT/CN2006/001550

INTERNATIONAL APPLICATION NUMBER

国际申请日: 03. JUL 2006 (03. 07. 2006)

INTERNATIONAL FILING DATE

明 名 称 : METHOD AND APPARATUS FOR FAST AUDIO SEARCH

OF INVENTION

CERTIFIED COPY OF
PRIORITY DOCUMENT

中华人民共和国国家知识产权局局长
COMMISSIONER OF THE STATE INTELLECTUAL PROPERTY
OFFICE OF THE PEOPLE'S REPUBLIC OF CHINA

司力善

二零零六年七月二十八日

JULY 28, 2006

BEST AVAILABLE COPY

HOME COPY

PCT

REQUEST

The undersigned requests that the present international application be processed according to the Patent Cooperation Treaty.

For receiving Office use only	
PCT/CN 2006 / 001550	
International Application No.	
03. III. 2006 (03.07.2006)	
International Filing Date	
RO/CN 中华人民共和国国家知识产权局	
Name of receiving Office and International Application	
Applicant's or agent's file reference (if desired) (12 characters maximum) FPEL06150037	

Box No. I TITLE OF INVENTION METHOD AND APPARATUS FOR FAST AUDIO SEARCH	
Box No. II APPLICANT	<input type="checkbox"/> This person is also inventor
Name and address: (Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant's State (that is, country) of residence if no State of residence is indicated below.)	
INTEL CORPORATION 2200 Mission College Blvd. Santa Clara, California 95052 United States of America	
<input type="checkbox"/> Telephone No.	
<input type="checkbox"/> Facsimile No.	
<input type="checkbox"/> Teleprinter No.	
<input type="checkbox"/> Applicant's registration No. with the Office	
State (that is, country) of nationality: US	State (that is, country) of residence: US
This person is applicant for the purposes of: <input type="checkbox"/> all designated States <input checked="" type="checkbox"/> all designated States except the United States of America <input type="checkbox"/> the United States of America only <input type="checkbox"/> the States indicated in the Supplemental Box	
Box No. III FURTHER APPLICANT(S) AND/OR (FURTHER) INVENTOR(S)	
Name and address: (Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant's State (that is, country) of residence if no State of residence is indicated below.)	
CHEN, Yurong Room 503, Unit 1, Building 4 QingFengHuaJing Yuan Haidian District, Beijing 100083 P. R. of China	
<input type="checkbox"/> This person is: applicant only	
<input checked="" type="checkbox"/> applicant and inventor	
<input type="checkbox"/> inventor only (If this check-box is marked, do not fill in below.)	
<input type="checkbox"/> Applicant's registration No. with the Office	
State (that is, country) of nationality: CN	State (that is, country) of residence: CN
This person is applicant for the purposes of: <input type="checkbox"/> all designated States <input type="checkbox"/> all designated States except the United States of America <input checked="" type="checkbox"/> the United States of America only <input type="checkbox"/> the States indicated in the Supplemental Box	
<input checked="" type="checkbox"/> Further applicants and/or (further) inventors are indicated on a continuation sheet.	
Box No. IV AGENT OR COMMON REPRESENTATIVE; OR ADDRESS FOR CORRESPONDENCE	
The person identified below is hereby/has been appointed to act on behalf of the applicant(s) before the competent International Authorities as:	
<input type="checkbox"/> agent <input type="checkbox"/> common representative	
Name and address: (Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)	
China Patent Agent (H.K.) Ltd. 22/F, Great Eagle Centre 23 Harbour Road, Wanchai Hong Kong Special Administrative Region The People's Republic of China	
<input type="checkbox"/> Telephone No. (852)28284688	
<input type="checkbox"/> Facsimile No. (852)28271018	
<input type="checkbox"/> Teleprinter No.	
<input type="checkbox"/> Agent's registration No. with the Office	
<input type="checkbox"/> Address for correspondence: Mark this check-box where no agent or common representative is/has been appointed and the space above is used instead to indicate a special address to which correspondence should be sent.	

Box No. V DESIGNATIONS

The filing of this request constitutes under Rule 4.9(a), the designation of all Contracting States bound by the PCT on the international filing date, for the grant of every kind of protection available and, where applicable, for the grant of both regional and national patents. However,

DE Germany is not designated for any kind of national protection
 JP Japan is not designated for any kind of national protection
 KR Republic of Korea is not designated for any kind of national protection
 RU Russian Federation is not designated for any kind of national protection

(The check-boxes above may only be used to exclude (irrevocably) the designations concerned if, at the time of filing, the international application contains in Box No. VI a priority claim to an earlier national application filed in the particular State concerned, in order to avoid the ceasing of the effect, under the national law, of this earlier national application. See the Notes to Box No. V as to the consequences of such national law provisions in these States).

Box No. VI PRIORITY CLAIM

The priority of the following earlier application(s) is hereby claimed:

Filing date of earlier application (day/month/year)	Number of earlier application	Where earlier application is:		
		national application: country or Member of WTO	regional application: regional Office	international application: receiving Office
item (1)				
item (2)				
item (3)				

Further priority claims are indicated in the Supplemental Box.

The receiving Office is requested to prepare and transmit to the International Bureau a certified copy of the earlier application(s) (only if the earlier application was filed with the Office which for the purposes of this international application is the receiving Office) identified above as:

all items item (1) item (2) item (3) other, see Supplemental Box

* Where the earlier application is an ARIPO application, indicate at least one country party to the Paris Convention for the Protection of Industrial Property or one Member of the World Trade Organization for which that earlier application was filed (Rule 4.10(b)(ii)):

Box No. VII INTERNATIONAL SEARCHING AUTHORITY

Choice of International Searching Authority (ISA) (if two or more International Searching Authorities are competent to carry out the international search, indicate the Authority chosen; the two-letter code may be used):

ISA /

Request to use results of earlier search; reference to that search (if an earlier search has been carried out by or requested from the International Searching Authority):

Date (day/month/year) Number Country (or regional Office)

Box No. VIII DECLARATIONS

The following declarations are contained in Boxes Nos. VIII (i) to (v) (mark the applicable check-boxes below and indicate in the right column the number of each type of declaration):

Number of
declarations

<input type="checkbox"/> Box No. VIII (i)	Declaration as to the identity of the inventor	:
<input type="checkbox"/> Box No. VIII (ii)	Declaration as to the applicant's entitlement, as at the international filing date, to apply for and be granted a patent	:
<input type="checkbox"/> Box No. VIII (iii)	Declaration as to the applicant's entitlement, as at the international filing date, to claim the priority of the earlier application	:
<input type="checkbox"/> Box No. VIII (iv)	Declaration of inventorship (only for the purposes of the designation of the United States of America)	:
<input type="checkbox"/> Box No. VIII (v)	Declaration as to non-prejudicial disclosures or exceptions to lack of novelty	:

Box No. IX CHECK LIST; LANGUAGE OF FILING

This international application contains:		This international application is accompanied by the following item(s) (mark the applicable check-boxes below and indicate in right column the number of each item):		Number of items
(a) on paper, the following number of sheets: request (including declaration sheets) : 3 description (excluding sequence listing and/or tables related thereto) : 24 claims : 8 abstract : 1 drawings : 8		1. <input checked="" type="checkbox"/> fee calculation sheet 2. <input checked="" type="checkbox"/> original separate power of attorney 3. <input type="checkbox"/> original general power of attorney 4. <input type="checkbox"/> copy of general power of attorney; reference number, if any: 5. <input type="checkbox"/> statement explaining lack of signature 6. <input type="checkbox"/> priority document(s) identified in Box No. VI as item(s): 7. <input type="checkbox"/> translation of international application into (language): 8. <input type="checkbox"/> separate indications concerning deposited microorganism or other biological material 9. <input type="checkbox"/> sequence listing in electronic form (indicate type and number of carriers)		
Sub-total number of sheets : 44		(i) <input type="checkbox"/> copy submitted for the purposes of international search under Rule 13ter only (and not as part of the international application): (ii) <input type="checkbox"/> (only where check-box (b)(i) or (c)(i) is marked in left column) additional copies including, where applicable, the copy for the purposes of international search under Rule 13ter (iii) <input type="checkbox"/> together with relevant statement as to the identity of the copy or copies with the sequence listing mentioned in left column		
Total number of sheets : 44		10. <input type="checkbox"/> tables in electronic form related to sequence listing (indicate type and number of carriers)		
(b) <input type="checkbox"/> only in electronic form (Section 801(a)(i)) (i) <input type="checkbox"/> sequence listing (ii) <input type="checkbox"/> tables related thereto		(i) <input type="checkbox"/> copy submitted for the purposes of international search under Section 802(b-quater) only (and not as part of the international application) (ii) <input type="checkbox"/> (only where check-box (b)(ii) or (c)(ii) is marked in left column) additional copies including, where applicable, the copy for the purposes of international search under Section 802(b-quater) (iii) <input type="checkbox"/> together with relevant statement as to the identity of the copy or copies with the tables mentioned in left column		
(c) <input type="checkbox"/> also in electronic form (Section 801(a)(ii)) (i) <input type="checkbox"/> sequence listing (ii) <input type="checkbox"/> tables related thereto		11. <input type="checkbox"/> other (specify): EN		
Type and number of carriers (diskette, CD-ROM, CD-R or other) on which are contained the				
<input type="checkbox"/> sequence listing: <input type="checkbox"/> tables related thereto: <i>(additional copies to be indicated under items 9(ii) and/or 10(ii), in right column)</i>				

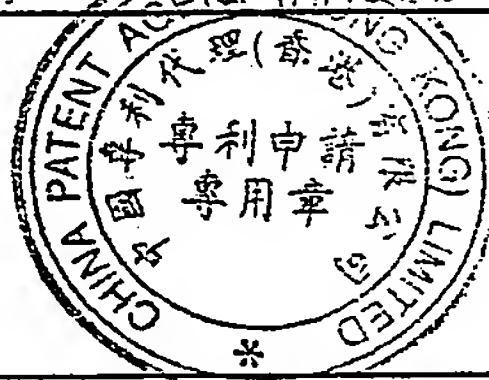
Figure of the drawings which should accompany the abstract:

Language of filing of the international application:

EN

Box No. X SIGNATURE OF APPLICANT, AGENT OR COMMON REPRESENTATIVE

Next to each signature, indicate the name of the person signing and the capacity in which the person signs (if such capacity is not obvious from reading the request).



For receiving Office use only

1. Date of actual receipt of the purported international application:	03 . JUL 2006 (03 . 07 . 2006)	2. Drawings:
3. Corrected date of actual receipt due to later but timely received papers or drawings completing the purported international application:		<input type="checkbox"/> received: <input type="checkbox"/> not received:
4. Date of timely receipt of the required corrections under PCT Article 11(2):		
5. International Searching Authority (if two or more are competent): ISA /	6. <input type="checkbox"/> Transmittal of search copy delayed until search fee is paid	

For International Bureau use only

Date of receipt of the record copy by the International Bureau:

This sheet is not part of and does not count as a sheet of the international application.

PCT

FEE CALCULATION SHEET
Annex to the Request

For receiving Office use only

PCT/CHN/2006/001550

International Application No.

03 · JUL 2006 (03 · 07 · 2006)

Date stamp of the receiving Office

Applicant's or agent's
file reference

FPEL06150037

Applicant

INTEL CORPORATION etc.

CALCULATION OF PRESCRIBED FEES

1. TRANSMITTAL FEE

CNY500

T

CNY500

2. SEARCH FEE

CNY2100

S

CNY 2100

International search to be carried out by

CN

(If two or more International Searching Authorities are competent to carry out the international search, indicate the name of the Authority which is chosen to carry out the international search.)

3. INTERNATIONAL FILING FEE

Where items (b) and/or (c) of Box No. IX apply, enter Sub-total number of sheets } 44
Where items (b) and (c) of Box No. IX do not apply, enter Total number of sheets }

i1 first 30 sheets

CHF1400

i1

CHF1400

i2 14 x CHF15 = CHF210
number of sheets in excess of 30

fee per sheet

CHF210

i2

CHF210

i3 additional component (only if sequence listing and/or tables related thereto are filed in computer readable form under Section 801(a)(i), or both in that form and on paper, under Section 801(a)(ii)):

400 x

fee per sheet

= i3

CHF1610

i

CHF1610

Add amounts entered at i1, i2 and i3 and enter total at I

(Applicants from certain States are entitled to a reduction of 75% of the international filing fee. Where the applicant is (or all applicants are) so entitled, the total to be entered at I is 25% of the international filing fee.)

4. FEE FOR PRIORITY DOCUMENT (if applicable)

P

CNY2600

5. TOTAL FEES PAYABLE

CNY2600CHF1610

TOTAL

CHF1610

MODE OF PAYMENT

authorization to charge
deposit account (see below)

postal money order

cash

coupons

cheque

bank draft

revenue stamps

other (specify):

AUTHORIZATION TO CHARGE (OR CREDIT) DEPOSIT ACCOUNT

(This mode of payment may not be available at all receiving Offices)

Authorization to charge the total fees indicated above.

(This check-box may be marked only if the conditions for deposit accounts of the receiving Office so permit) Authorization to charge any deficiency or credit any overpayment in the total fees indicated above.

Authorization to charge the fee for priority document.

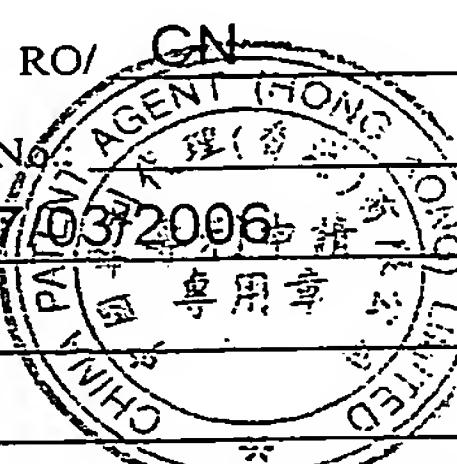
Receiving Office: RO/ CN

Deposit Account No.: AGENT (HONG KONG)

Date: 07/03/2006

Name: (Signature)

Signature: (Signature)



See Notes to the fee calculation sheet

METHOD AND APPARATUS FOR FAST AUDIO SEARCH

BACKGROUND

5 1. FIELD

This disclosure relates generally to signal processing and multimedia applications, and more specifically but not exclusively, to methods and apparatus for fast audio search and audio fingerprinting.

10 2. DESCRIPTION

Audio search (e.g., searching a large audio stream for an audio clip, even if the large audio stream is corrupted/distorted) has many applications including analysis of broadcast music/commercials, copyright management over the Internet, or finding metadata for unlabeled audio clips, and etc. A typical audio 15 search system is serial and designed for single processor systems. It normally takes a long time for such a search system to search for a target audio clip in a large audio stream. In many cases, however, an audio search system is required to work efficiently on large audio databases, e.g., to search large databases in a very short time (e.g., close to real-time). Additionally, an audio 20 database may be partially or entirely distorted, corrupted, and/or compressed. This requires that an audio search system be robust enough to identify those audio segments that are the same as the target audio clip, even if those segments may be distorted, corrupted, and/or compressed. Thus, it is desirable

to have an audio search system which can quickly and robustly search large audio databases for a target audio clip.

BRIEF DESCRIPTION OF THE DRAWINGS

5 The features and advantages of the disclosed subject matter will become apparent from the following detailed description of the subject matter in which:

Figure 1 shows one example computing system where robust and parallel audio search may be performed using an audio search module;

10 Figure 2 shows another example computing system where robust and parallel audio search may be performed using an audio search module;

Figure 3 shows yet another example computing system where robust and parallel audio search may be performed using an audio search module;

Figure 4 is a block diagram of an example audio search module that performs robust audio search;

15 Figure 5 is an example illustrating how a robust audio search module shown in Figure 4 works;

Figure 6 is a block diagram of an example audio search module that performs robust and parallel audio search in a multiprocessor system;

20 Figures 7A, 7B, and 7C illustrate a method of partitioning a large audio database into smaller groups for robust and parallel audio search in a multiprocessor system; and

Figure 8 is pseudo code illustrating an example process for performing robust and parallel audio search in a multiprocessor system.

DETAILED DESCRIPTION

According to embodiments of the subject matter disclosed in this application, a large audio stream or a large audio database in a multiprocessor system may be searched for a target audio clip using a robust and parallel

5 search method. The large audio database may be partitioned into a number of smaller groups. These smaller groups may be dynamically scheduled to be processed by available processors or processing cores in the multiprocessor system. Processors or processing cores may process the scheduled groups in parallel by partitioning each group into smaller segments, extracting acoustic
10 features from the segments; and modeling the segments using a common component Gaussian Mixture model ("CCGMM"). The length of these segments may be the same as the length of the target audio clip. Before processing any group, one processor or processing core may extract acoustic features from the target audio clip and model it using the CCGMM. A Kullback-Leibler (KL) or KL-
15 max distance may be further computed between the model of the target audio clip and each segment of a group. If the distance equals or smaller than a predetermined value, the corresponding segment is identified as the target audio clip.

If the distance is larger than a predetermined value, the processor or
20 processing core may skip a certain number of segments and continue searching for the target audio clip. Once a processor or processing core finishes searching a group, a new groups may be given to it for processing to search for the target audio clip until all of the groups are searched. The size of the groups may be

determined in such a way to reduce the load imbalance and the overlapped computation. Furthermore, Input/Output (I/O) may be optimized to improve the efficiency of parallel processing of audio groups by multiple processors or processing cores.

5 Reference in the specification to "one embodiment" or "an embodiment" of the disclosed subject matter means that a particular feature, structure or characteristic described in connection with the embodiment is included in at least one embodiment of the disclosed subject matter. Thus, the appearances of the phrase "in one embodiment" appearing in various places throughout the 10 specification are not necessarily all referring to the same embodiment.

Figure 1 shows one example computing system 100 where robust and parallel audio search may be performed using an audio search module 120. Computing system 100 may comprise one or more processors 110 coupled to a system interconnect 115. Processor 110 may have multiple or many processing 15 cores (for brevity of description, term "multiple cores" will be used hereinafter to include both multiple processing cores and many processing cores). Processor 110 may include an audio search module 120 to conduct robust and parallel audio search by multiple cores. The audio search module may comprise several components such as a partitioning mechanism, a schedule, and multiple audio 20 searchers (see more detailed description for Figures 4-6 below). One or more components of the audio search module may be located in one core with others in another core.

The audio search module may first partition a large audio database into multiple smaller groups or a large audio stream into smaller partially overlapped substreams. Second, one core may process an audio clip to be searched for ("target audio clip") to establish a model for the target audio clip. In the mean 5 while, the audio search module dynamically schedules smaller audio groups/substreams to multiple cores, which partition each group/substream into segments and establish a model for each audio segment, in parallel. The size of each segment may be equal to the size of the target audio clip. A Gaussian mixture model ("GMM") with multiple Gaussian components, which are common 10 to all of the audio segments including both the target audio clip and the audio database/stream, may be used for modeling each audio segment and the target audio clip. Once a model is established for an audio segment, Kullback-Leibler 15 ("KL") or KL-max distance may be computed between the segment model and the target audio clip model. If the distance is not larger than a predetermined value, the audio segment may be identified as the target audio clip. The search 20 process may continue until all audio groups/substreams are processed.

The computing system 100 may also include a chipset 130 coupled to the system interconnect 115. Chipset 130 may include one or more integrated circuit packages or chips. Chipset 130 may comprise one or more device 20 interfaces 135 to support data transfers to and/or from other components 160 of the computing system 100 such as, for example, BIOS firmware, keyboards, mice, storage devices, network interfaces, etc. Chipset 130 may be coupled to a Peripheral Component Interconnect (PCI) bus 170. Chipset 130 may include a

PCI bridge 145 that provides an interface to the PCI bus 170. The PCI Bridge 145 may provide a data path between the processor 110 as well as other components 160, and peripheral devices such as, for example, an audio device 180 and a disk drive 190. Although not shown, other devices may also be 5 coupled to the PCI bus 170.

Additionally, chipset 130 may comprise a memory controller 125 that is coupled to a main memory 150. The main memory 150 may store data and sequences of instructions that are executed by multiple cores of the processor 110 or any other device included in the system. The memory controller 125 may 10 access the main memory 150 in response to memory transactions associated with multiple cores of the processor 110, and other devices in the computing system 100. In one embodiment, memory controller 150 may be located in processor 110 or some other circuitries. The main memory 150 may comprise various memory devices that provide addressable storage locations which the 15 memory controller 125 may read data from and/or write data to. The main memory 150 may comprise one or more different types of memory devices such as Dynamic Random Access Memory (DRAM) devices, Synchronous DRAM (SDRAM) devices, Double Data Rate (DDR) SDRAM devices, or other memory devices.

20 Figure 2 shows another example computing system 200 where robust and parallel audio search may be performed using an audio search module 240. System 200 may comprise multiple processors such as processor 220A. One or more processors in system 200 may have many cores. System 200 may

include an audio search module 240 to conduct robust and parallel audio search by multiple cores. The audio search module may comprise several components such as a partitioning mechanism, a schedule, and multiple audio searchers (see more detailed description for Figures 4-6 below). One or more components 5 of the audio search module may be located in one core with others in another core. Processors in system 200 may be connected to each other using a system interconnect 210. System interconnect 210 may be a Front Side Bus (FSB). Each processor may be connected to Input/Output (IO) devices as well as memory 230 through the system interconnect. All of the cores may receive 10 audio data from memory 230.

Figure 3 shows yet another example computing system 300 where robust and parallel audio search may be performed using an audio search module 340. In system 300, system interconnect 310 that connects multiple processors (e.g., 320A, 320B, 320C, and 320D) is a links-based point-to-point connection. Each 15 processor may connect to the system interconnect through a links hub (e.g., 330A, 330B, 330C, and 330D). In some embodiments, a links hub may be co-located with a memory controller, which coordinates traffic to/from a system memory. One or more processor may have many cores. System 300 may include an audio search module 340 to conduct robust and parallel audio search 20 by multiple cores. The audio search module may comprise several components such as a partitioning mechanism, a schedule, and multiple audio searchers (see more detailed description for Figures 4-6 below). One or more components of the audio search module may be located in one core with others in another

core. Each processor/core in system 300 may be connected to a shared memory (not shown in the figure) through the system interconnect. All of the cores may receive audio data from the shared memory.

In Figures 2 and 3, the audio search module (i.e., 240 and 340) may first

- 5 partition a large audio database into multiple smaller groups or a large audio stream into smaller partially overlapped substreams. Second, one core may process an audio clip to be searched for ("target audio clip") to establish a model for the target audio clip. In the mean while, the audio search module dynamically schedules smaller audio groups/substreams to multiple cores, which
- 10 partition each group/substream into segments and establish a model for each audio segment, in parallel. The size of each segment may be equal to the size of the target audio clip. A Gaussian mixture model ("GMM") with multiple Gaussian components, which are common to all of the audio segments including both the target audio clip and the audio database/stream, may be used for
- 15 modeling each audio segment and the target audio clip. Once a model is established for an audio segment, Kullback-Leibler ("KL") or KL-max distance may be computed between the segment model and the target audio clip model. If the distance is not larger than a predetermined value, the audio segment may be identified as the target audio clip. The search process may continue until all
- 20 audio groups/substreams are processed.

Figure 4 is a block diagram of an example audio search module 400 that performs robust audio search. Audio search module 400 comprises a feature extractor 410, a modeling mechanism 420, and a decision maker 430. Feature

extractor 410 may receive an input audio stream (e.g., a target audio clip, a substream of a large audio stream, etc.) and extract acoustic features from the input audio stream. When the input audio stream is an audio stream to be searched for the target audio clip, the feature extractor may apply sliding window

5 on the audio stream to partition it into multiple overlapped segments. The window has the same length as the target audio clip. Each segment of the input audio stream (the target audio stream has only one segment) is further separated into frames. Each frame may have the same length and may overlap with its neighboring frames. For example, in one embodiment, a frame may be
10 20 milliseconds in length with the overlap between frames being 10 milliseconds. A feature vector may be extracted for each frame, which may include such features as Fourier coefficients, Mel-Frequency cepstral coefficients, spectral flatness, and means, variances, other derivatives thereof. Feature vectors from all of the frames in an audio segment form a feature vector sequence.

15 The overlap between two adjacent segments is to reduce the likelihood of missing any target audio clip between two adjacent segments. The longer the overlap is, the less likely a miss is. In one embodiment, the overlap may be equal to the length of a segment minus the length of a frame to avoid missing any match. However, longer overlap means more computation. Thus, there
20 should be a balance between the computation load and the likelihood of miss (e.g., the overlap is equal to or less than $\frac{1}{2}$ of the segment length). In any case, feature vectors for frames that are overlapped between two segments only need to be extracted once.

Modeling mechanism 420 may establish a model for an audio segment based on its feature vector sequence extracted by feature extractor 410. Depending on what model is used, the modeling mechanism will estimate parameters for the model. In one embodiment, a common component Gaussian mixture model ("CCGMM") may be used for modeling an audio segment. The CCGMM includes multiple Gaussian components which are common across all of the segments. For each segment, the modeling mechanism estimates a specific set of mixture weights for the common Gaussian components. In another embodiment, other models (e.g., hidden Markov model) may be used for modeling an audio segment. In one embodiment, only the target audio clip may be modeled; and the feature vector sequence of an audio segment may be directly used to determine whether the audio segment is substantially the same as the target audio clip.

Decision maker 430 may determine whether an audio segment in the input audio stream is sufficiently similar so that the audio segment can be identified as a copy of the target audio clip. To achieve this goal, the decision maker may derive a similarity measure by comparing the model of the audio segment and the model of the target audio clip. In one embodiment, the similarity measure may be a distance computed between the two models. In another embodiment, the similarity measure may be probability of the audio segment model being the same as the target audio clip model. Yet in another embodiment, the similarity measure may be derived by comparing the feature vector sequence of the audio segment and the model of the target audio clip.

For example, when a hidden Markov model ("HMM") is used to model the target audio clip, a Viterbi based algorithm may be used to compute a likelihood score between the audio segment and the target audio clip, based on the feature vector sequence of the audio segment and the HMM of the target audio clip.

5 Based on the value of the similarity measure, the decision maker may determine whether an audio segment can be identified as the target audio clip. For example, if the value of the similarity measure is not larger than a predetermined threshold (e.g., similarity measure is distance between the audio segment model and the target audio clip), the audio segment may be identified 10 as substantially the same as the target audio clip. Similarly, the audio segment may be identified as substantially the same as the target audio clip if the value of the similarity measure is not smaller than a predetermined threshold (e.g., similarity measure is a likelihood score of the audio segment being substantially the same as the target audio clip). On the other hand, if an audio segment is 15 found to be substantially different from the target audio clip based on the similarity measure, a certain number of segments immediately following the audio segment may be skipped. The actual number of segments to be skipped will depend on the value of the similarity measure and/or empirical data. By skipping a number of following segments, it is not likely to miss any target audio 20 clip when the similarity measure indicate the current segment is so different from the target audio clip because the window used to partition an input audio stream into segments slides forward gradually and as a result, there is continuity of similarity measure from one segment to the next.

Figure 5 is an example illustrating how a robust audio search module shown in Figure 4 works. A target audio clip 510 is received by a feature extractor which segments it into frames and produces a feature vector sequence (540) at block 530A, with a feature vector per frame. A feature vector may be an

5 x dimensional vector (wherein $x \geq 1$) because the feature vector may include one or more parameters. At block 570A, Feature vector sequence 540 may be modeled using a GMM as shown below:

$$P^{(k)}(x) = \sum_{i=1}^M w_i^{(k)} N(x | \mu_i^{(k)}, \sum_i^{(k)}). \quad (1)$$

The GMM, $P^{(k)}(x)$, includes M Gaussian components with component 10 weights $w_i^{(k)}$, means $\mu_i^{(k)}$, and covariance $\sum_i^{(k)}$, with $i = 1, 2, \dots, M$; wherein k denotes segment k and $N()$ denotes a Gaussian distribution. For the target audio clip, there is only one segment, and hence there is no need to use k to identify a segment. For the input audio stream 520, however, there is typically more than one segment, and it is thus desirable to identify the GMM for different 15 segments.

In the example shown in Figure 5, Kullback-Leibler (KL) or KL-max distance is used as a similarity measure. To simplify KL-max distance computation, it is assumed that the GMMs used for all the audio segments share a common set of Gaussian components, i.e., for the i^{th} Gaussian component, the 20 mean (μ_i) and variance (\sum_i) are the same across different audio segments.

As a result, Equation (1) becomes:

$$P^{(k)}(x) = \sum_{i=1}^M w_i^{(k)} N(x | \mu_i, \sum_i). \quad (2)$$

For each audio segment, only a set of weights, $W_i^{(k)}, i = 1, 2, \dots, M$, need to be estimated for the common Gaussian components. Given a feature vector sequence for segment k , which has T feature vectors, $x_t (t = 1, 2, \dots, T)$, weights

5 may be estimated as follows,

$$w_i^{(k)} = \frac{1}{T} \sum_{t=1}^T \frac{w_i^{(u)} N(x_t | \mu_i, \sum_i)}{\sum_{j=1}^M w_j^{(u)} N(x_t | \mu_j, \sum_j)}, \quad (3)$$

wherein $w_i^{(u)}$ or $w_j^{(u)}$ is a universal weight for the i^{th} or j^{th} segment, which may be obtained by experiments based on some sample audio files or be initialized with a random value.

10 An input audio stream 520, which is to be searched for the target audio clip 510, may be received by a feature extractor. At block 530B, the feature extractor partitions the input audio stream into partially overlapped segments. For each segment, the feature extractor further partitions the segment into multiple partially overlapped frames and extracts a feature vector from each
 15 frame. Block 560 shows a feature vector sequence for the input audio stream 520 and also illustrates how the audio stream is partitioned into partially overlapped segments. For example, a window with the size being the same as the length of the target audio clip may be applied to input audio stream 520. For illustration purpose, a window is shown for the feature vector sequence of the
 20 target audio clip to obtain a segment 560A although there is typically no need to apply a window to the target audio clip since there is only one segment. A

shifting window is applied to the input audio stream to obtain multiple partially overlapped segments such as 560B and 560C. The window shifts by time τ from segment 560B to segment 560C, where τ is smaller than the window size.

Each audio segment is modeled using the CCGMM, for example, segment

5 560B is modeled at block 570B and segment 560C is modeled at block 570C.

Models for each segment of input audio stream 520 and for target audio clip 510

have common Gaussian components with different sets of weights. In one

embodiment, feature vectors may be extracted from the entire input audio stream

frame by frame to produce a long feature vector sequence for the entire input

10 audio stream. A window with a length being $N \times FL$ (where N is a positive integer

and FL is the frame length) is subsequently applied to the long feature vector

sequence. Feature vectors within the window constitute a feature vector for an

audio segment, which is used to establish a CCGMM. The window is shifting

forward by τ time.

15 To determine if a segment is substantially the same as the target audio

clip, KL-max distance may be calculated between the model of the segment and

the target audio clip as follows,

$$d_{KLMAX} = \max_{i=1,2,\dots,M} (w_i^{(1)} - w_i^{(2)}) \log \frac{w_i^{(1)}}{w_i^{(2)}}. \quad (4)$$

If the KL-max distance so calculated is below a predetermined threshold, the

20 audio clip may be considered to be detected. As the window applied over input

audio stream 520 shifts forward in time, distances typically show certain

continuity from one time-step to the next. In other words, if the distance is too

large, it is unlikely that one or more segments immediately following the current segment matches the target audio clip. Thus, depending on the value of the distance, a certain number of immediately following segments in the same audio stream/substream may be skipped from search.

5 Figure 6 is a block diagram of an example audio search module 600 that performs robust and parallel audio search in a multiprocessor system. The audio search module 600 comprises a partitioning mechanism 610, a scheduler 620, an I/O optimizer 630, and a plurality of audio searchers (e.g., 640A, ..., 640N). Partitioning mechanism 610 may partition a large audio stream into 10 multiple smaller substreams and/or a large audio database into multiple smaller groups. Figures 7A, 7B, and 7C illustrate a method of partitioning a large audio database into smaller groups for robust and parallel audio search in a multiprocessor system. Figure 7A shows an example database that contains a single large audio stream 710. The partitioning mechanism may partition audio 15 stream 710 into multiple smaller substreams such as 712, 714, and 716, with each substream constituting a group. The length of substreams can vary from each other, but it is normally uniform for the simplicity purpose. To avoid missing any correct detection of a target audio clip, each substream overlaps 20 with its immediately following substream; and the overlap between two adjacent substreams (e.g., 712 and 714, 714 and 716) should equal or longer than $FNClip - 1$, where $FNClip$ is the total number of frames in the target audio clip.

Figure 7B shows another example database that includes multiple relatively small audio streams (e.g., 720, 725; 730, 735, and 740). In one

embodiment, partitioning mechanism 610 may partition the database into multiple smaller groups with each group consisting of only one audio stream. In another embodiment, the partitioning mechanism may partition the database into multiple smaller groups with some groups each consisting of only one audio

5 stream and with others each consisting of more than one small audio stream, as illustrated in Figure 7B. Figure 7C shows yet another example database that includes some relatively small audio streams (e.g., 750, 755, and 760) as well as some large audio stream (e.g., 770). The partitioning mechanism may put those relatively small audio streams into groups with each group consisting of only one 10 audio stream or with some groups consisting of only one audio stream (e.g., 750) while others consisting of more than one small audio streams (e.g., 755 and 760 may be grouped together). As for a large audio stream such as 770, the partitioning mechanism may partition it into multiple partially overlapped smaller substreams (e.g., 712 and 714) with each substream constituting a group, using 15 the method illustrated in Figure 7A.

Additionally, the partitioning mechanism partitions a large audio database into groups with proper sizes to reduce the overlapped computation (in the situation where a large audio stream is partitioned into multiple overlapped smaller substreams) and load imbalance in parallel processing by multiple 20 processors. Smaller group size may result in large overlapped computation, while larger group size may result in considerable load imbalance. In one embodiment, the group size may be about 25 times of the size of the target audio clip.

Turning back to Figure 6, scheduler 620 may dynamically schedule multiple groups of a large database into multiple processors in the multiprocessor system with each processor having one group to process at one time. The scheduler periodically checks the availability of processors in the system and assigns an audio group for each available processor to process and search for the target audio clip. If another processor becomes available later, the scheduler may assign one group to this processor. The scheduler also assigns an unsearched audio group to processor immediately after it finishes searching its previously assigned group no matter whether other processors finish their searching. In fact, even for groups with the same size, searching for the same target audio clip may take different amount of time for different processors because the number of segments to be skipped may be different from one segment to another. Using dynamic scheduling as outlined above may further reduce load imbalance effectively.

I/O optimizer 630 may optimize I/O traffic on the system interconnect (e.g., system bus connecting a shared system memory with processors in the system). The I/O optimizer may decide not to load the entire audio database to be searched for from the disk into the memory in the beginning while the data range for each processor is being defined. Additionally, the I/O optimizer may let each processor read only a portion of its assigned segment from the memory at one time. By optimizing the I/O traffic, the I/O optimizer may reduce I/O contention, implement the overlap of I/O operations and computation, and help to improve

computation efficiency. As a result, the scalability of audio search can be significantly improved.

Audio search module Audio 600 also comprises a plurality of audio searchers 640A through 640N. Each audio searcher (e.g., 640A) is located in a

5 processor to process a group assigned to the processor and to search for the target audio clip. Similar to an audio searching module 400 shown in Figure 4, an audio searcher includes a feature extractor (e.g., 410), a modeling mechanism (e.g., 420), and a decision maker (e.g., 430). Each audio searcher conducts serial active search of an audio group assigned to it for a target audio
10 clip by partitioning audio streams in the audio group into partially overlapped segments with length being the same as the target audio clip, extracting feature vector sequence for each segment, and modeling each segment using a CCGMM as illustrated in Equations (1) through (4). Additionally, the CCGMM for the target audio clip which is used by all of the audio searchers just needs to
15 be estimated once by one of the audio searchers. Each audio searcher computes KL-max distance between the model for each segment and the model of the target audio clip. Based on the KL-max distance, an audio searcher may determine if the target audio clip is detected. Moreover, each audio searcher may skip a number of segments that following the current segment if the KL-max
20 distance for the current segment is larger than a threshold.

Figure 8 is pseudo code illustrating an example process 800 for performing robust and parallel audio search in a multiprocessor system. At line 802, audio search module may be initialized, e.g., target audio clip file and audio

database file may be opened, and global parameters may be initialized. At line 804, a large audio database may be partitioned into NG smaller groups as illustrated in Figures 7A, 7B, and 7C. At line 806, a model (e.g., CCGMM) may be established for the target audio clip. At line 808, NG audio groups may be

5 dynamically scheduled to available processors and parallel processing of the scheduled groups may be started. Line 808 uses one example instruction that sets up parallel implementation and other parallel implementation instructions may also be used.

Lines 810 through 846 illustrate how each of NG groups are processed 10 and searched for the target in parallel by a processor in the multiprocessor system. It is worth noting that for illustration purpose, process in lines 812 to 846 is shown as iteration from the first group until the last group. In practice, if there are several processors available, several groups are processed in parallel by these available processors. At line 814, some or all of audio streams in each 15 group may be further partitioned into NS partially overlapped segments if such streams are longer in time than the target audio clip. Line 816 starts iterative process for each segment of the group, shown in lines 818 through 832. At line 820, a feature vector sequence (frame by frame) may be extracted from the segment. At line 822, a model (e.g., CCGMM as shown in Equations (1) to (3)) 20 may be established for the segment. At line 824, distance (e.g., KL-max distance as shown in Equation (4)) between the segment model and the target audio clip model may be computed. At line 826, whether the segment matches the target audio clip or not may be determined based on the distance calculated

in line 824 and a predetermined threshold #1. If the distance is less than the threshold #1, the segment matches the target audio clip. At line 828, whether a number of following segments (e.g., M segments) in the same audio stream/substream may be skipped from searching may be determined based on

- 5 the distance calculated in line 824 and a predetermined threshold #2. If the distance is larger than the threshold #2, M segments may be skipped from searching. In one embodiment, the number of segments to be skipped may vary depending upon the value of the distance. At line 830, the search results (e.g., index or starting time of a match segment in each group) may be stored in an array which is local to the processor that processes the group. At line 842, search results from local arrays from all of the processors may be summarized and outputted to a user.

Using the robust and parallel search strategy as outlined in Figure 8 along with other techniques such as I/O optimization, search speed for a target 15 audio clip in a large audio database in a multiprocessor system may be significantly improved. One experiment shows that search speed for a 15 second target audio clip in a 27 hour audio stream increases by 11 times on a 16-way Unisys system, compared to serial search of the same audio stream for the same target audio clip.

20 In one embodiment, a modified search strategy may be used. Using this strategy, a preliminary model (e.g., CCGMM) may be established for the first K frames ($K \geq 1$) of the target audio clip along with a full model for the entire target audio clip. Accordingly, a preliminary model (e.g., CCGMM) may be first

established for the first K frames ($K \geq 1$) of an audio segment. During active search, the preliminary model of the first K frames of each audio segment may be first compared with the preliminary model of the first K frames of the target audio clip to produce a preliminary similarity measure. If the preliminary 5 similarity measure indicates that these two preliminary models are significantly similar, a full model may be established for the entire audio segment and compared with the full model of the entire target audio clip; otherwise, no full model will be established for the audio segment and the next segment may be searched by first establishing a preliminary model for its first K frames and by 10 comparing this preliminary model with the preliminary model of the target audio clip. This modified search strategy may further reduce computation load.

Although an example embodiment of the disclosed subject matter is described with reference to block and flow diagrams in Figures 1-8, persons of ordinary skill in the art will readily appreciate that many other methods of 15 implementing the disclosed subject matter may alternatively be used. For example, the order of execution of the blocks in flow diagrams may be changed, and/or some of the blocks in block/flow diagrams described may be changed, eliminated, or combined.

In the preceding description, various aspects of the disclosed subject 20 matter have been described. For purposes of explanation, specific numbers, systems and configurations were set forth in order to provide a thorough understanding of the subject matter. However, it is apparent to one skilled in the art having the benefit of this disclosure that the subject matter may be practiced

without the specific details. In other instances, well-known features, components, or modules were omitted, simplified, combined, or split in order not to obscure the disclosed subject matter.

Various embodiments of the disclosed subject matter may be

5 implemented in hardware, firmware, software, or combination thereof, and may be described by reference to or in conjunction with program code, such as instructions, functions, procedures, data structures, logic, application programs, design representations or formats for simulation, emulation, and fabrication of a design, which when accessed by a machine results in the machine performing
10 tasks, defining abstract data types or low-level hardware contexts, or producing a result.

For simulations, program code may represent hardware using a hardware description language or another functional description language which essentially provides a model of how designed hardware is expected to perform.

15 Program code may be assembly or machine language, or data that may be compiled and/or interpreted. Furthermore, it is common in the art to speak of software, in one form or another as taking an action or causing a result. Such expressions are merely a shorthand way of stating execution of program code by a processing system which causes a processor to perform an action or produce
20 a result.

Program code may be stored in, for example, volatile and/or non-volatile memory, such as storage devices and/or an associated machine readable or machine accessible medium including solid-state memory, hard-drives, floppy-

disks, optical storage, tapes, flash memory, memory sticks, digital video disks, digital versatile discs (DVDs), etc., as well as more exotic mediums such as machine-accessible biological state preserving storage. A machine readable medium may include any mechanism for storing, transmitting, or receiving

5 information in a form readable by a machine, and the medium may include a tangible medium through which electrical, optical, acoustical or other form of propagated signals or carrier wave encoding the program code may pass, such as antennas, optical fibers, communications interfaces, etc. Program code may be transmitted in the form of packets, serial data, parallel data, propagated
10 signals, etc., and may be used in a compressed or encrypted format.

Program code may be implemented in programs executing on programmable machines such as mobile or stationary computers, personal digital assistants, set top boxes, cellular telephones and pagers, and other electronic devices, each including a processor, volatile and/or non-volatile
15 memory readable by the processor, at least one input device and/or one or more output devices. Program code may be applied to the data entered using the input device to perform the described embodiments and to generate output information. The output information may be applied to one or more output devices. One of ordinary skill in the art may appreciate that embodiments of the
20 disclosed subject matter can be practiced with various computer system configurations, including multiprocessor or multiple-core processor systems, minicomputers, mainframe computers, as well as pervasive or miniature computers or processors that may be embedded into virtually any device.

Embodiments of the disclosed subject matter can also be practiced in distributed computing environments where tasks may be performed by remote processing devices that are linked through a communications network.

Although operations may be described as a sequential process, some of

5 the operations may in fact be performed in parallel, concurrently, and/or in a distributed environment, and with program code stored locally and/or remotely for access by single or multi-processor machines. In addition, in some embodiments the order of operations may be rearranged without departing from the spirit of the disclosed subject matter. Program code may be used by or in
10 conjunction with embedded controllers.

While the disclosed subject matter has been described with reference to illustrative embodiments, this description is not intended to be construed in a limiting sense. Various modifications of the illustrative embodiments, as well as other embodiments of the subject matter, which are apparent to persons skilled 15 in the art to which the disclosed subject matter pertains are deemed to lie within the scope of the disclosed subject matter.

CLAIMS

What is claimed is:

1. A method for searching an audio database for a target audio clip in a multiprocessor system, comprising:
 2. partitioning said audio database into a plurality of groups;
 3. establishing a model for said target audio clip;
 4. dynamically scheduling said plurality of groups to a plurality of processors in said multiprocessor system; and
 5. processing said scheduled groups in parallel by said plurality of processors to search for said target audio clip.
1. 2. The method of claim 1, wherein partitioning said audio database comprises determining a size for each of said plurality of groups, said size being determined to reduce the amount of overlapped computation among said plurality of groups and load imbalance in parallel processing of said plurality of groups.
1. 3. The method of claim 1, wherein establishing a model for said target audio clip comprises extracting a feature vector sequence from said target audio clip and modeling said feature vector sequence based on a Gaussian Mixture model ("GMM"), said GMM including a plurality of Gaussian components.

1 4. The method of claim 3, wherein modeling said feature vector sequence
2 comprises estimating mixture weights for each of said plurality of Gaussian
3 components.

1 5. The method of claim 1, wherein processing said scheduled groups in
2 parallel comprises:
3 partitioning each of said scheduled groups into at least one segment; and
4 for each segment,

5 extracting a feature vector sequence for the segment, and
6 modeling said feature vector sequence based on a Gaussian
7 Mixture model ("GMM"), said GMM including a plurality of Gaussian
8 components.

1 6. The method of claim 5, wherein each of said at least one segment has
2 the same length in time as that of said target audio clip.

1 7. The method of claim 5, wherein if there are more than one segments in
2 an audio stream, each segment partially overlaps with a segment that
3 immediately precedes that segment.

1 8. The method of claim 5, wherein said plurality of Gaussian components
2 are common for different segments and said target audio clip.

1 9. The method of claim 8, wherein modeling said feature vector sequence
2 comprises estimating mixture weights for each of said plurality of Gaussian
3 components.

1 10. The method of claim 9, further comprising: for each segment,
2 computing a Kullback-Leibler ("KL") distance between a GMM of said
3 segment and a GMM of said target audio clip; and
4 determining that said segment matches said target audio clip, if said KL
5 distance is smaller than a pre-determined threshold.

1 11. The method of claim 10, further comprising skipping processing a
2 number of segments if said KL distance is larger than a predetermined value,
3 said number of segments dependent on the value of said KL distance.

1 12. The method of claim 1, wherein said multiprocessor system
2 comprises a memory shared by said plurality of processors.

1 13. An apparatus for searching an audio database for a target audio clip
2 in a multiprocessor system, comprising:
3 a partitioning module to partition said audio database into a plurality of
4 groups;
5 a scheduler to dynamically schedule said plurality of groups to a plurality
6 of processors in said multiprocessor system; and

7 an audio searching module for each of said plurality of processors to
8 process said scheduled groups in parallel by said plurality of processors to
9 search for said target audio clip.

1 14. The apparatus of claim 13, wherein said partitioning module further
2 determines a size for each of said plurality of groups, said size being determined
3 to reduce the amount of overlapped computation among said plurality of groups
4 and load imbalance in parallel processing of said plurality of groups.

1 15. The apparatus of claim 13, wherein an audio searching module
2 comprises:
3 a feature extractor to partition an input audio stream into at least one
4 segment and to extract a feature vector sequence from each of said at least one
5 segment, said at least one segment having the same length in time as that of
6 said target audio clip; and
7 a modeling module to model said feature vector sequence for each
8 segment based on a Gaussian Mixture model ("GMM"), said GMM including a
9 plurality of Gaussian components, said plurality of Gaussian components being
10 common among all of the segments.

1 16. The apparatus of claim 15, wherein one of audio searching modules
2 further process said target audio clip by extracting a feature vector sequence
3 from said target audio clip and by modeling said feature vector sequence using

4 said GMM, said GMM including a plurality of Gaussian components common for
5 said target audio clip and segments of said input audio stream.

1 17. The apparatus of claim 16, wherein an audio searching module
2 further comprising a decision maker to compute a Kullback-Leibler ("KL")
3 distance between a GMM of a segment of said input audio stream and a GMM of
4 said target audio clip; and to determine whether said segment matches said
5 target audio clip based on said KL distance.

1 18. The apparatus of claim 17, wherein said decision module further
2 determines how many segments are to be skipped from processing based on
3 said KL distance.

1 19. An article comprising a machine-readable medium that contains
2 instructions, which when executed by a processing platform, cause said
3 processing platform to perform operations comprising:
4 partitioning said audio database into a plurality of groups;
5 establishing a model for said target audio clip;
6 dynamically scheduling said plurality of groups to a plurality of processors
7 in said multiprocessor system; and
8 processing said scheduled groups in parallel by said plurality of
9 processors to search for said target audio clip.

1 20. The article of claim 19, wherein partitioning said audio database
2 comprises determining a size for each of said plurality of groups, said size being
3 determined to reduce the amount of overlapped computation among said
4 plurality of groups and load imbalance in parallel processing of said plurality of
5 groups.

1 21. The article of claim 19, wherein establishing a model for said target
2 audio clip comprises extracting a feature vector sequence from said target audio
3 clip and modeling said feature vector sequence based on a Gaussian Mixture
4 model ("GMM"), said GMM including a plurality of Gaussian components.

1 22. The article of claim 21, wherein modeling said feature vector
2 sequence comprises estimating mixture weights for each of said plurality of
3 Gaussian components.

1 23. The article of claim 19, wherein processing said scheduled groups in
2 parallel comprises:
3 partitioning each of said scheduled groups into at least one segment; and
4 for each segment,
5 extracting a feature vector sequence for the segment, and
6 modeling said feature vector sequence based on a Gaussian
7 Mixture model ("GMM"), said GMM including a plurality of Gaussian
8 components.

1 24. The article of claim 22, wherein each of said at least one segment
2 has the same length in time as that of said target audio clip.

1 25. The article of claim 22, wherein if there are more than one segments
2 in an audio stream, each segment partially overlaps with a segment that
3 immediately precedes that segment.

1 26. The article of claim 22, wherein said plurality of Gaussian components
2 are common for different segments and said target audio clip.

1 27. The article of claim 26, wherein modeling said feature vector
2 sequence comprises estimating mixture weights for each of said plurality of
3 Gaussian components.

1 28. The article of claim 27, wherein said operations further comprise: for
2 each segment,
3 computing a Kullback-Leibler ("KL") distance between a GMM of said
4 segment and a GMM of said target audio clip; and
5 determining that said segment matches said target audio clip, if said KL
6 distance is smaller than a pre-determined threshold.

1 29. The article of claim 28, wherein said operations further comprise
2 skipping processing a number of segments if said KL distance is larger than a
3 predetermined value, said number of segments dependent on the value of said
4 KL distance.

1 30. The article of claim 19, wherein said multiprocessor system
2 comprises a memory shared by said plurality of processors.

ABSTRACT OF THE DISCLOSURE

According to embodiments of the subject matter disclosed in this application, a large audio database in a multiprocessor system may be searched for a target audio clip using a robust and parallel search method. The large audio database may be partitioned into a number of smaller groups, which are dynamically scheduled to available processors in the system. Processors may process the scheduled groups in parallel by partitioning each group into smaller segments, extracting acoustic features from the segments; and modeling the segments using a common component Gaussian Mixture model ("CCGMM"). One processor may also extract acoustic features from the target audio clip and model it using the CCGMM. Kullback-Leibler (KL) distance may be further computed between the target audio clip and each segment. Based on the KL distance, a segment may be determined to match the target audio clip; and/or a number of following segments may be skipped.

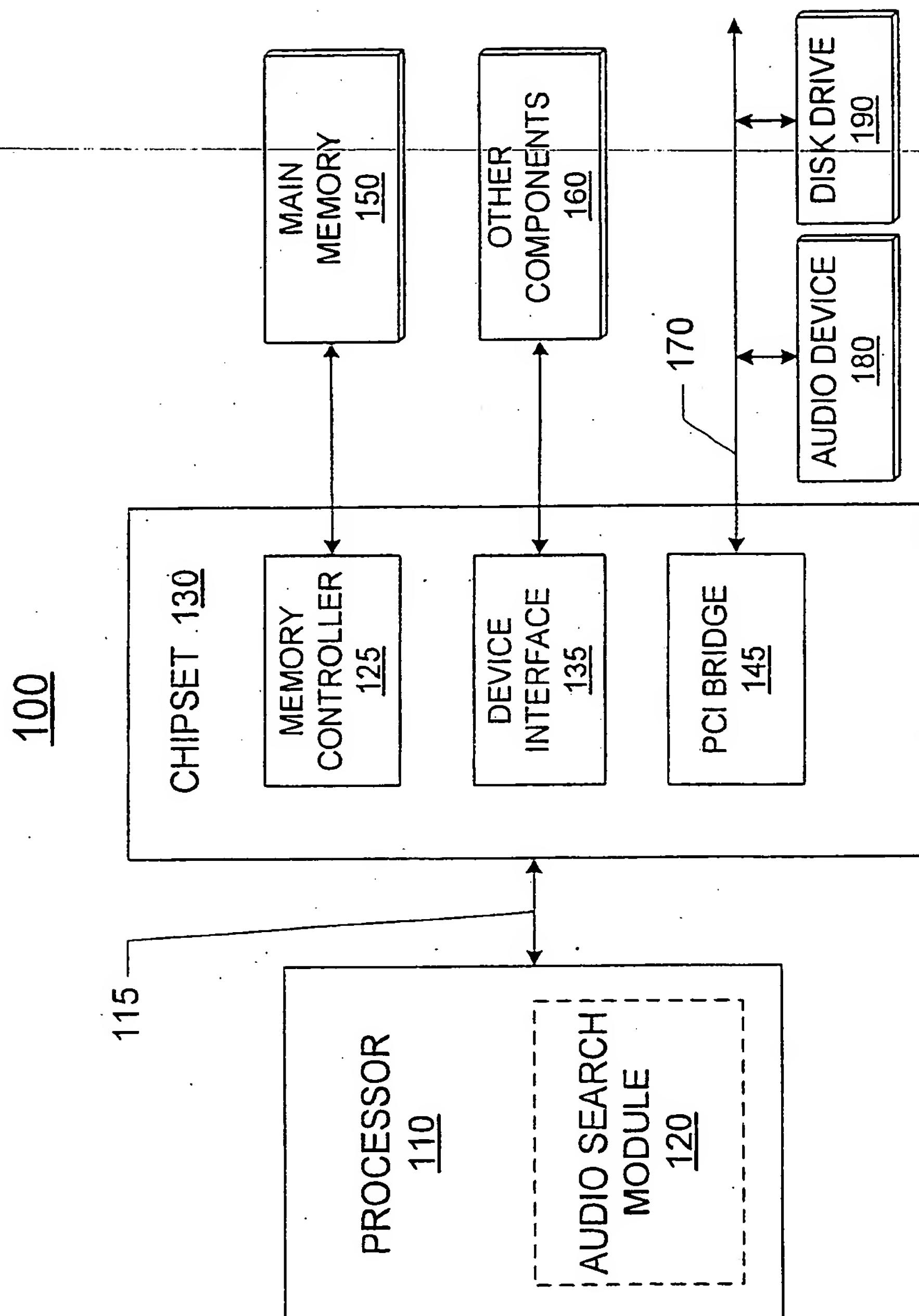


FIGURE 1

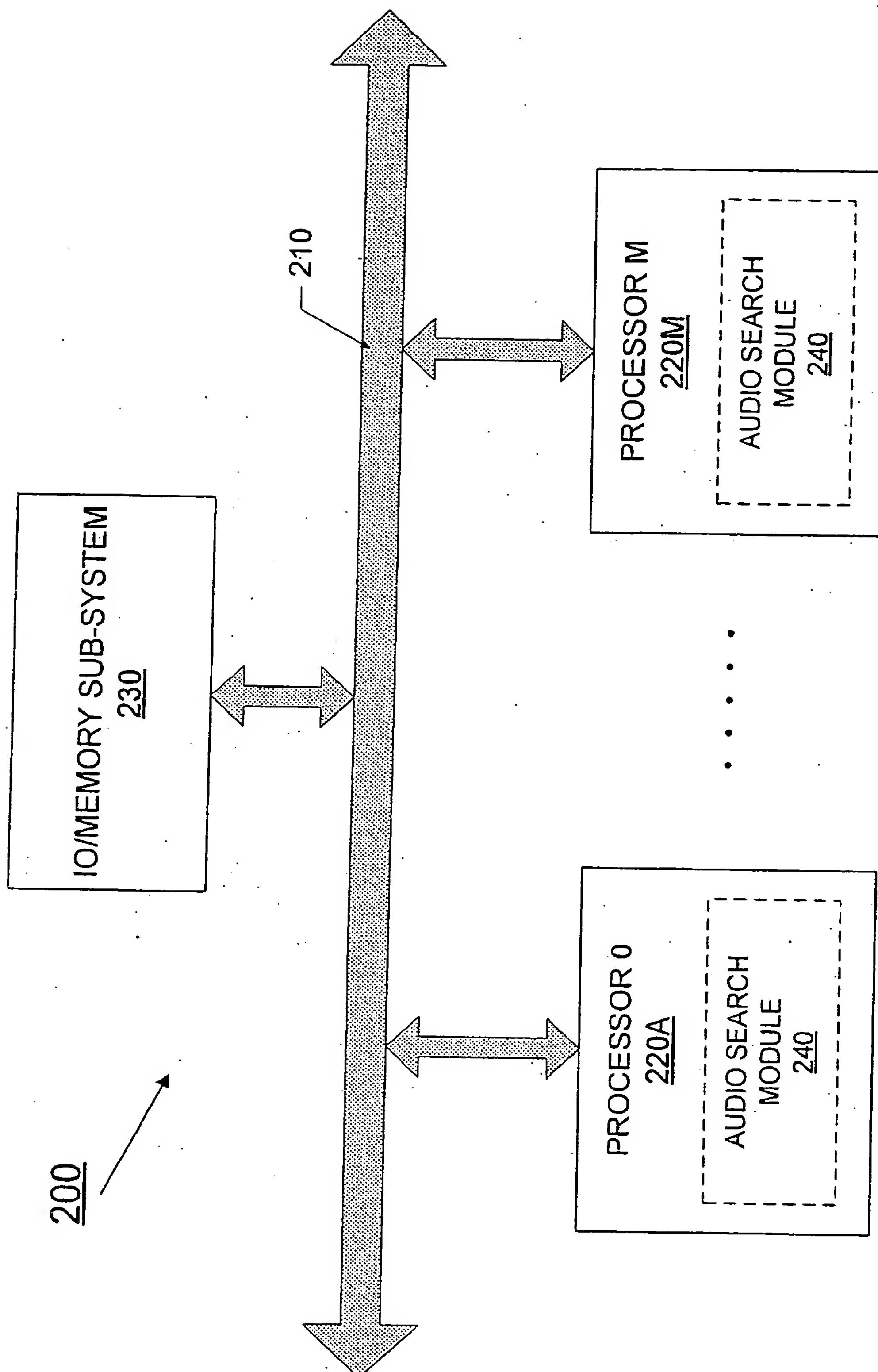


FIGURE 2

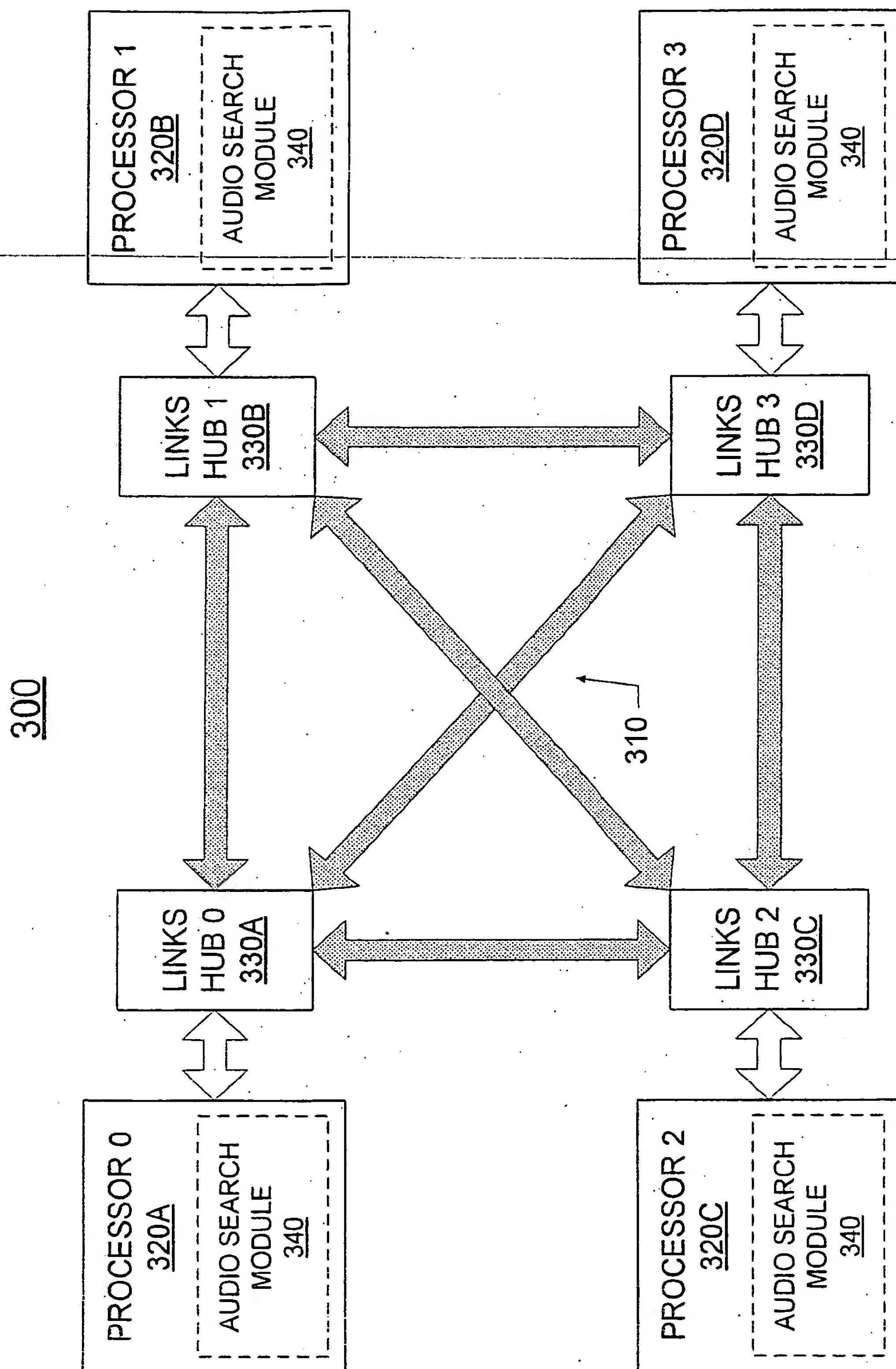


FIGURE 3

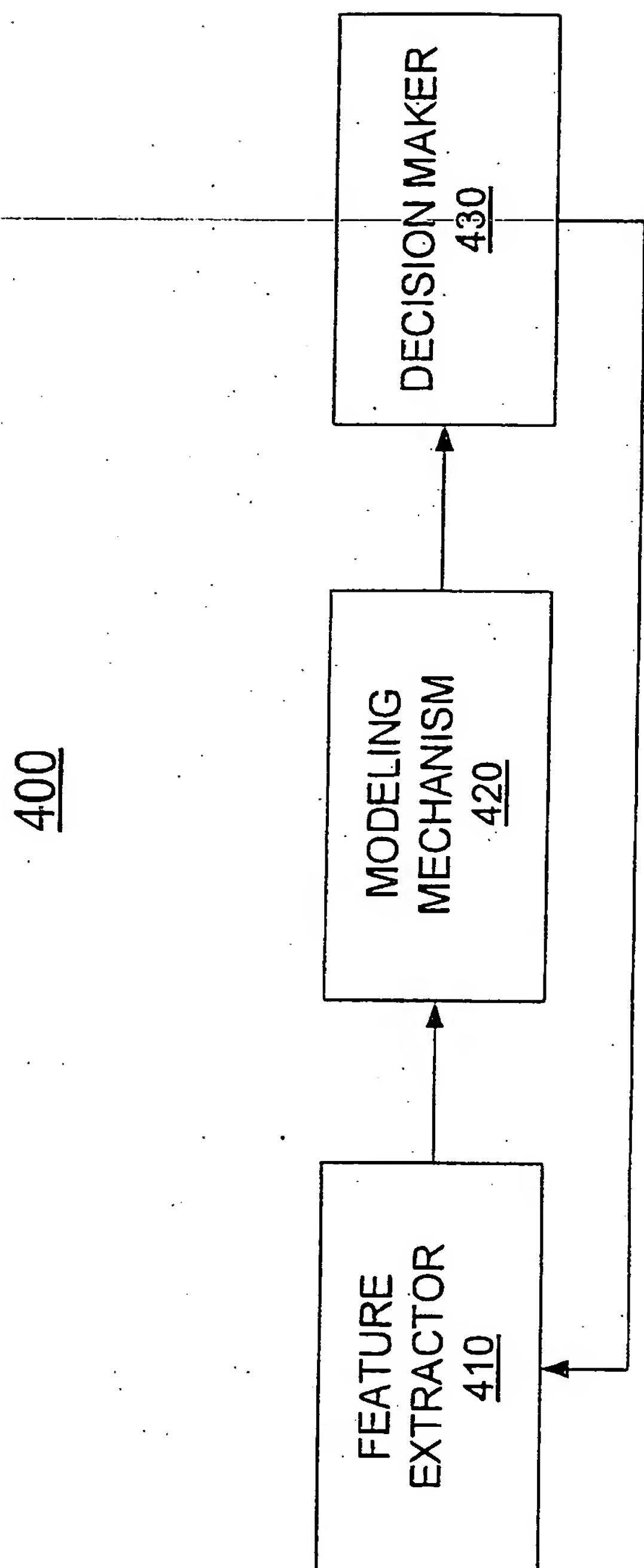


FIGURE 4

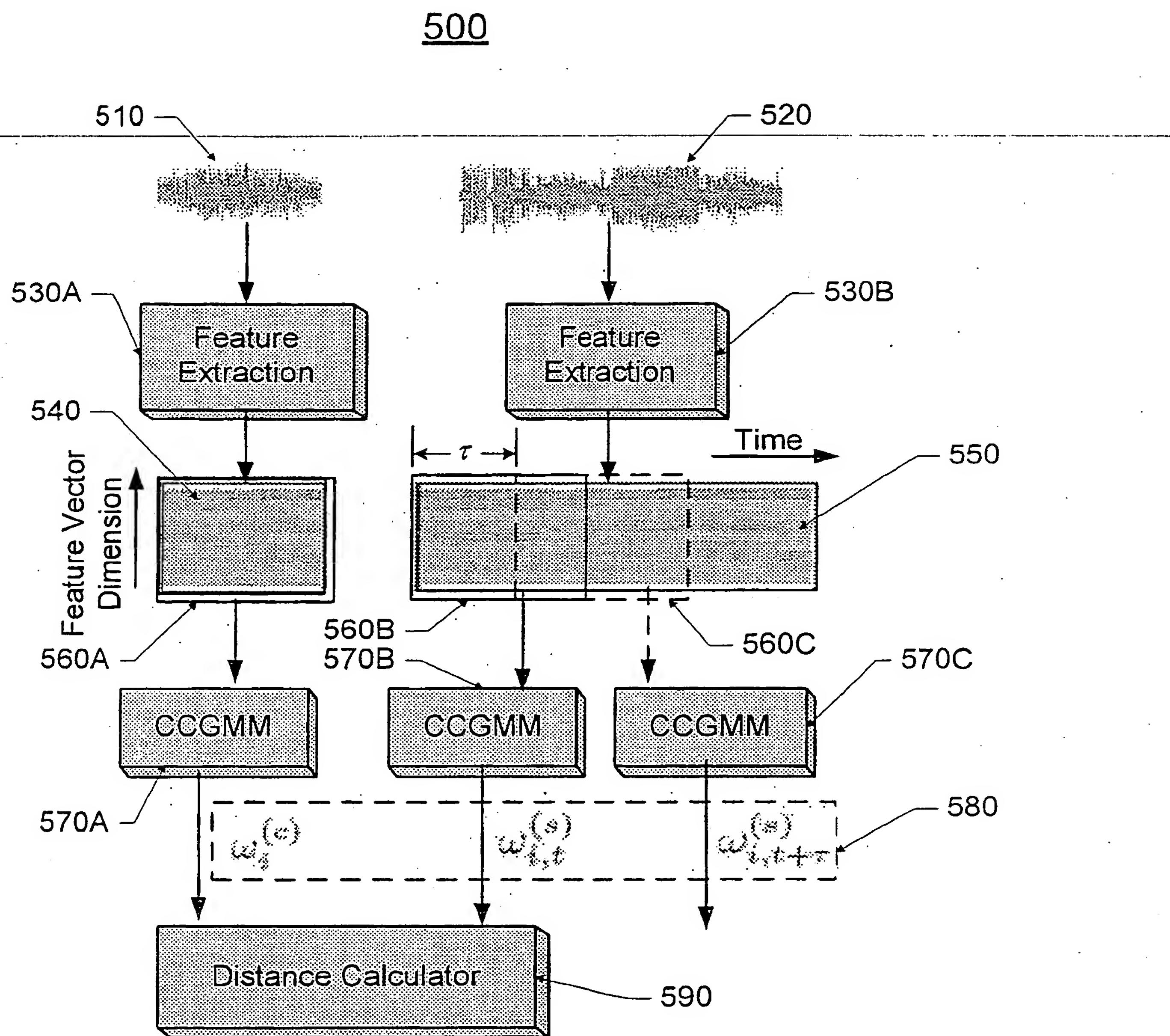


FIGURE 5

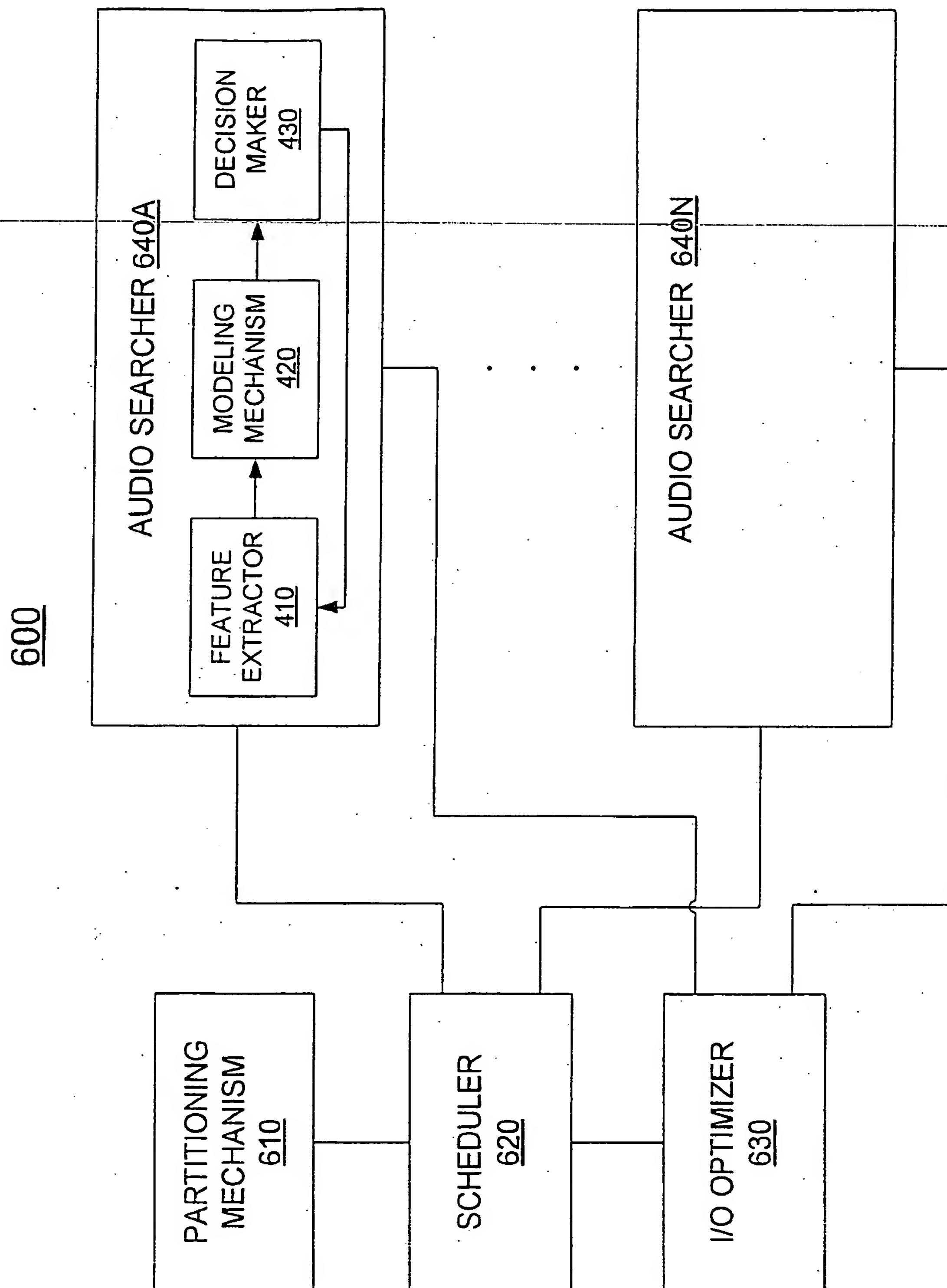


FIGURE 6

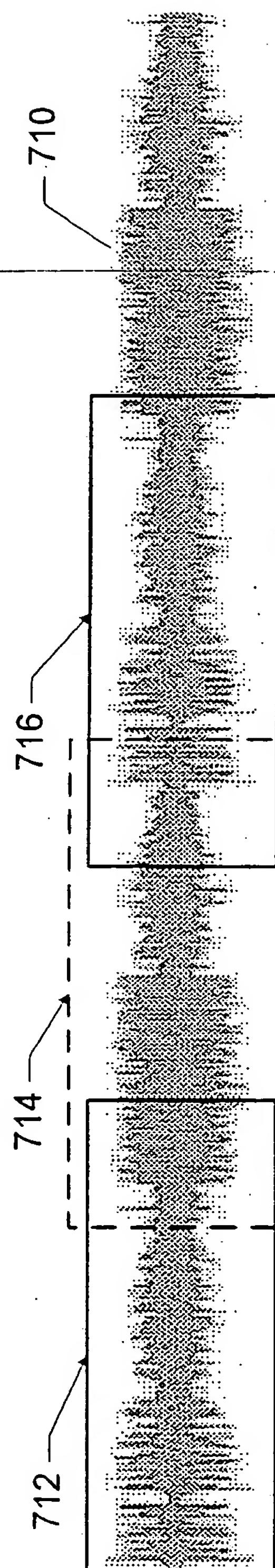


FIGURE 7A

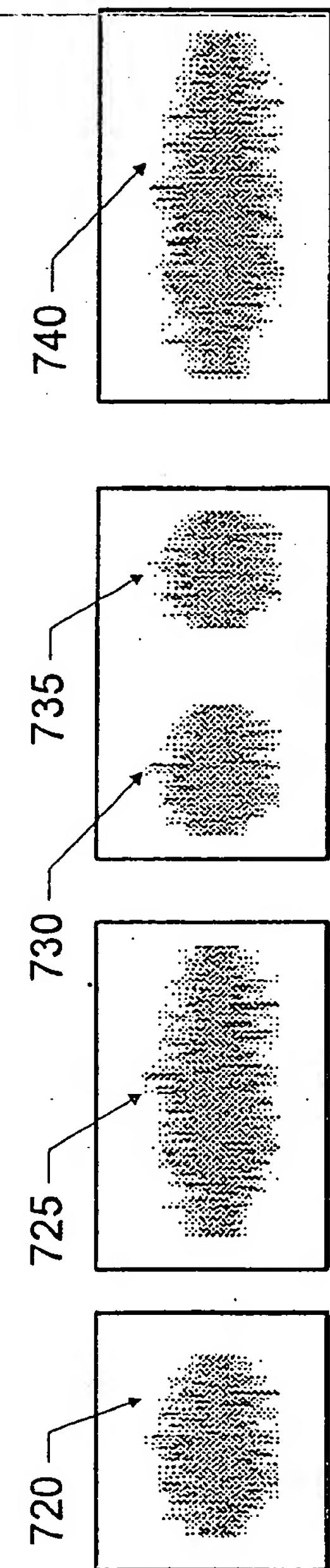


FIGURE 7B

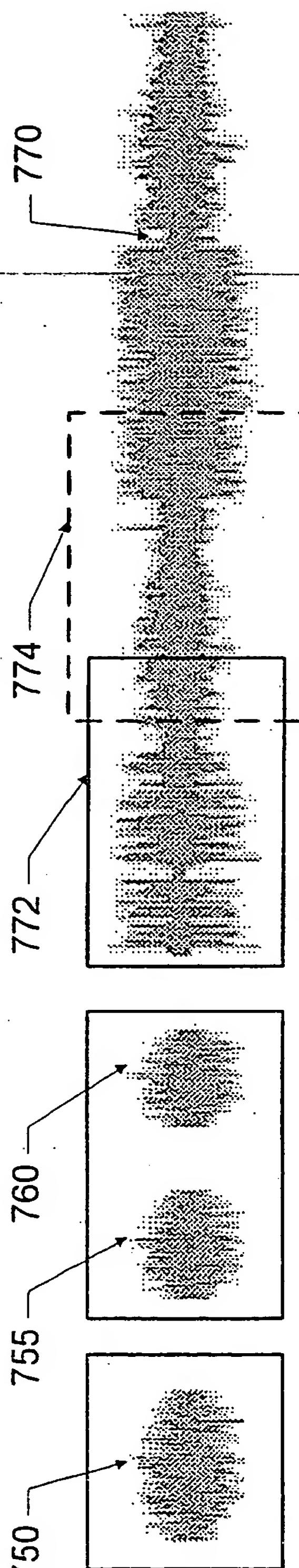


FIGURE 7C

800

```
802: Initialization;
804: Partition a large audio database into NG smaller groups;
806: Establish a model for target audio clip;
808: #pragma omp parallel for schedule(dynamic,1),
      num_threads(NumOfThread);
      /* dynamically schedule smaller groups to available processors and
      start parallel processing of the scheduled groups by multiple
      processors */
810: For groupid = 0 to NG-1
812: {
814:     Partition current group into NS partially overlapped
          segments, if necessary;
816:     For segmentid = 0 to NS-1
818:     {
820:         Extract a feature vector sequence;
822:         Establish a model for the segment;
824:         Compute distance between the model of each
              segment and the target audio clip model;
826:         If Distance < threshold #1, Match!
828:         else if Distance > threshold #2,
              Skip M segments in the same audio stream;
830:         Store results into an local array for the group;
832:     }
834: }
842: Output search results of local arrays from each processor ;
```

FIGURE 8

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.